Inter-Transaction Association Rules Mining for Rare Events Prediction

Christos Berberidis, Lefteris Angelis and Ioannis Vlahavas

Department of Informatics, Aristotle University of Thessaloniki,
54124 Thessaloniki, Greece
{berber, lef, vlahavas}@csd.auth.gr

Abstract. Rare events prediction is a very interesting and critical issue that has been approached within various contexts by research areas, such as statistics and machine learning. Data mining has provided a set of tools to treat this problem when the size as well as the inherent features of the data, such as noise, randomness and special data types, become an issue for the traditional methods. Transaction databases that contain sets of events require special approaches in order to extract valuable temporal knowledge. Sequential analysis aims to discover patterns or rules describing the temporal structure of data. In this paper we propose an approach that extends sequential analysis to predict rare events in transaction databases. We utilize the framework of inter-transaction association rules, which associate events across a window of transactions. The proposed algorithm produces rules for the accurate and timely prediction of a user-specified rare event, such as a network intrusion or an engine failure.

Keywords: Data Mining, Rule Learning, Prediction.

1. Introduction

Data mining has drawn the attention of the research and business sector for more than a decade now, since it has provided decision makers with a set of powerful tools to exploit very large amounts of data. Data mining is particularly interesting and useful because the miner is able to discover knowledge from data that have been collected for other purposes. What makes it even more challenging is that the extracted knowledge has to be understandable, useful and interesting and also has to be delivered in time. Data mining tools employ a number of methods mainly from statistics, artificial intelligence, machine learning and database technology, in order to produce predictive or descriptive models of the data. This paper proposes an approach to the interesting task of learning to predict rare events from transactional databases, such as rapid change in stock prices, engine failures, network intrusions or failures etc. For this purpose, we utilize the recently proposed framework of inter-transactional association rules.

A very important area in the data miming research is the mining of association rules, a very simple but useful form of rule patterns for knowledge discovery. Association rules were initially used as a tool to apply market basket analysis to sets of data but they were extended to other kinds of analyses, such as sequential pattern mining, generalization and prediction. The typical problem of mining association rules concerns the search of associations among discrete items in a database. The goal is to find all sets of items that occur frequently in the same transaction and from those sets to derive rules that one subset of an itemset implies another. The notion of transaction is very general and includes items bought by the same customer, requests from same user, events happened on the same day, etc. The traditional association analysis is intra-transactional because it concerns items within the same transaction. Inter-transactional association rules comprise a new kind of association rules that associate items within a window of many transactions. In that sense, intra-transactional rules are a subset of the inter-transactional ones.

Special types of association rules have been proposed for extracting time-related information from temporal data. The intra-transactional association rule mining algorithms were extended to capture sequences of events. Temporal association rules require a transaction database model, where items are grouped together into *transactions*. each transaction having the following form: *(Tid, Cid,* I_1*,* I_2*,…,* I_n*, t)*, where *Tid* is the ID of the transaction and *Cid* is the ID of the Customer. $I_{1,...,}In$ are the items contained in the transaction and *t* is a timestamp.

In classical association rules mining, records in a transactional database contain only items and are identified by their transaction Ids (TIDs). Although transactions occur under certain *contexts* such as time, place, customers, etc., such contextual information has been ignored because this task was intra-transactional. A classical association rule example could be the most commonly quoted: *"If a customer buys diapers then he also buys beer, with support s, confidence c"*. This rule associates items found in the same transaction (here, market basket), so these rules are intra-transactional. However, rules like *"If the prices of IBM and SUN go up, Microsoft's will most likely (80% of the time) goes up 2 days later"* cannot be captured with the approaches proposed so far. This kind of rule associates itemsets among different transactions, along the axis of time. The contextual information here is time, which is the *dimensional attribute*. One can discover rules along multiple dimensions, depending on the amount of contextual information in the data. For example, assume the following 2-dimensional rule: "*After McDonald and KFC open branches in this area, Burger King will also open a branch, 1 month later, within 1 mile away"*. In this case, the dimensional attributes are time and space. Those rules are called *inter-transactional* and they can be single or multi dimensional. The major advantage of intertransactional association rules is that besides description they can also facilitate prediction, providing the user with explicit dimensional (in the case of prediction, temporal) information. It is often useful to know when to expect something to happen with accuracy (e.g. "five days later") instead of a fuzzy temporal window (e.g. "some day *within 1 week*") or a sequence (e.g. A and B will happen *after* C).

In this paper we propose a method for predicting rare events in a transaction database. We utilize the predictive power of inter-transactional association rules to discover predictive patterns.

The paper is organized as follows: The next section presents a review of the literature regarding sequential analysis, rare events prediction and inter-transaction association rules. In section 3 we provide the definitions and theoretical background of our approach. The algorithm we propose is described in section 4, along with a brief discussion about its computational complexity and performance. In section 5 we present the experiments we conducted in order to test and verify the performance of the proposed algorithm and finally, in section 6 we present our conclusions and propose our ideas for further work.

2. Related Work

Event prediction is very similar to time series prediction. Classical time series prediction, which has been studied extensively within the field of statistics, involves predicting the next *n* successive observations from a history of past observations (Brockwell & Davis 1996). These statistical techniques involve the building of mathematical probabilistic models, which are based on specific data, since they are strongly dependent on various theoretical assumptions regarding the underlying nature of variation (probability distributions etc). However, this is not our case. First, we are interested in extracting knowledge from a very broad class of large transaction data bases, without any prior information on the variability of the data and therefore without having to state theoretical assumptions. Second, our main goal is not to build certain mathematical models, but to discover patterns and rules, which are related to certain critical events and which are going to provide us an alarm for the early identification of such events.

Having a temporal database, we can mine for various types of association rules. One approach is to cluster the data based on time and then discover association rules from each cluster, in order to track how the model changes over time [9]. Traditional association rule analysis was extended to *sequence analysis*, where the members of the series are sets of individual *items*, called *itemsets*, from some underlying domain (alphabet). Given a set E of events, an *event sequence* s is a sequence of pairs (e, t), where $e \in E$ and t is an integer, the occurrence time of the event of type e. Unlike time series, sequences do not require any explicit relationship with time, only that the itemsets are totally ordered.

The basic difference between the two concepts, then, is that a time series is a list of ordered values, while a sequence is a list of ordered itemsets or values. Sequential pattern mining aims to discover patterns such as {{A}, {B}, {C, D}}, where {A}, {B} and {C, D} are itemsets in different transactions, within a user-defined time window. Finding the most frequent maximal patterns is a particularly useful task that provides the user with valuable insight about the temporal nature of the data. However, the predictive power of sequential association rules is questionable. Sequence analysis or sequential pattern mining was extensively studied initially by Agrawal et al. [6, 7], where the notions of sequence and subsequence were defined.

An *episode rule* [8] is a generalization of association rules applied to sequences of events. An *event sequence S* is an ordered list of events, each one occurring at a particular time. Thus, it can be viewed as a special type of time series. Given the above

definitions, an *episode a* is a partial order of event types. Episodes can be viewed as directed acyclic graphs. There are serial, parallel and non-serial and non-parallel episodes. Episode mining algorithms are searching for episodes or episode rules within a sliding window of user-defined size. What is captured here is the temporal relationship among events that occur within the same window, e.g. "C comes after A and B within a window of size *w*".

Inter-transactional association rules were introduced in [1] and [2]. The authors extend the notion of inter-transactional association rules to the multidimensional space and propose EH-Apriori, an Apriori-based algorithm, for mining such rules. The authors also propose the use of templates and concept hierarchies as a means to reduce the large number of the produced rules. A new set of algorithms is introduced in [3], called FITI (an acronym for "First Intra then Inter"), featuring better performance than EH-Apriori. In [4] and [5], the authors use inter-transactional association rules for prediction on meteorological and stock market data, correspondingly.

3. Problem Formulation - Definitions

In our setup, we used notions defined in [1, 2, 12], adjusting and enriching them in order to provide an appropriate and complete theoretical and mathematical basis for the prediction of rare events.

- The set of *items* $I = \{i_1, i_2, \ldots, i_v\}$ representing the possible activities we want to keep record of (e.g. items sold in a store or responses to requests by a server).
- The *dimensional variable* T describing the time properties associated with the items. We assume that the variable takes ordinal values representing intervals of equal length (e.g. day, week, month etc.). Note that this variable can be defined to represent various other ordinal measurements such as length, height, etc. It is also possible to have many of these variables (time, distance, etc.), simultaneously describing our data, but in our context we consider only one. Without loss of generality we denote the values of T by integers 0, 1, 2, ...
- The *transactions* which are records of the form $J(t)$ where t is a value of the time variable T and $J(t) \subseteq I$. So, each transaction is represented by a set of certain activities from I recorded in time t.
- The *transaction database* containing all the transactions recorded over a (usually long) period of time.
- The *transaction sequence*, a time-ordered sequence of transactions, denoted by $S = J(t_1), J(t_2), \ldots, J(t_n)$, which includes n transactions recorded in the time interval $[t_1, t_n]$.
- The *target item*, $i^* \in I$, which represents an activity that we are particularly interested in predicting it, e.g. failure of a system to respond, network fault, etc. Such an item occurs infrequently with respect to the other items while its occurrence is

much more critical than the others'. Let us also denote by t^* the time interval when the target item occurs.

- The *target transactions* which are transactions containing the target item. Note that for our work here we do not need to consider target items at all but more generally target transactions, since a target transaction may have the meaning of an infrequent combination of items that we are interested to predict.

So, the problem we consider here is to derive inter-transactional association rules that can be used as alarm messages in order to predict the target transactions within a reasonable period of time before the critical target item occurs. For this purpose we associate with every target transaction $J(t^*)$:

- A *prediction period*, which is a time period preceding the target transaction of fixed length defined as $[t^* - m, t^* - w]$ where m is the *monitoring time* and w is the *warning time*. We assume that $m > w$.
- A *target - preceding window* W^*, which is a block of $m - w + 1$ continuous time intervals included in the prediction period of the target transaction. Thus, the window consists of all the time intervals from $t^* - m$ to $t^* - w$. Note that it is not necessary for each interval to contain a transaction. These intervals within a window are called *target - preceding subwindows* of W^*. We will use non-negative integers to denote the subwindows. So, we denote the subwindow in the beginning of the prediction period by $W^*(1)$, and the following ones by $W^*(1),\ldots,W^*(m-w+1)$. We will also use the same indices to denote the items in each subwindow. Thus, if the item i_k $(1 \le k \le v)$ occurs in *target - preceding subwindow* $W^*(x)$ $(1 \le x \le m - w + 1)$, it will be denoted by $i_k(x)$. Such items are called *extended items.* We denote the set of all possible extended items by
$$I^* = \{i_k(x): \quad 1 \le k \le v, \ \ 1 \le x \le m - w + 1\}.$$
- A *target megatransaction* $M^* \subseteq I^*$ defined as the set of all extended items within W^*, i.e.
$$M^* = \{i_k(x) : i_k \in W^*(x), \quad 1 \le k \le v, \quad 1 \le x \le m - w + 1\}.$$
- An *intertransaction association rule predicting the target transaction* is an implication of the form $F \Rightarrow G$ where $F \subseteq I^*$ and G is the target transaction.
- Two *measures* of intertransaction association rules $F \Rightarrow G$:
$$\textit{support of } F: \ s = \frac{N_F^*}{N^*} \qquad (1)$$
where N^* is the number of all the target megatransactions in the database and N_F^* is the number of all target megatransactions that contain the set F.

We can characterize a set as *frequent* if s exceeds a lower bound, defined by the user.

$$\text{confidence of } F: c = \frac{N_F^*}{N_F} \qquad (2)$$

where N_F is the set of all megatransactions of size $m-w+1$ in the database that contain the set F. We can characterize a set as *accurate* if c exceeds a lower bound, defined by the user.

The purpose of the search is to find all frequent and accurate sets of extended items, and then to find rules able to predict the target transactions.

4. The Prediction Algorithm

Our approach is based on the inter-transactional association rules framework, which provides an excellent basis for producing predictive rules. The general strategy we follow to predict rare events takes into account the fact that it is highly important that a prediction is given in time. Therefore, we assume that there is a time period preceding a target event X_t, when the prediction can be useful (prediction period or monitoring window). This period starts with a time point that denotes the beginning of the period when the user is interested in having a prediction and ends with a time point after which it is too late and the prediction has no meaning (warning time). The concept is illustrated below:

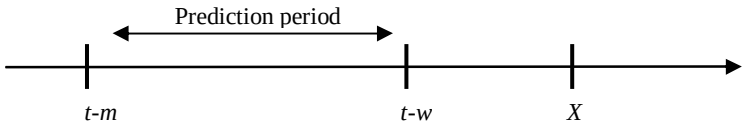

Fig. 1. The prediction period.

Given that the warning time is always *w* time points before the target event X_t, we propose an efficient method for mining predictive patterns within the prediction time period. We perform one scan over the database in order to capture and store only the transactions associated with those periods, using a sliding window. The number of such periods (windows) stored is equal to the number of occurrences of the target event, which, in our case is rare. In other words, we capture the corresponding monitoring window of every occurrence of the target event in order to extract the desired knowledge. While capturing those windows, a database transformation takes place in order to map the relative temporal information of every item within the window. The transformation is done according to the definitions given in the previous paragraph, as provided by the inter-transactional association rules framework. An example of such transformation can be seen in the following figure.

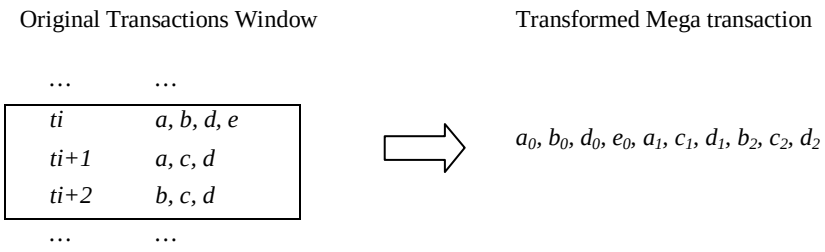

Fig. 2. A data transformation example.

For memory efficiency purposes, we map every item instance i_t to an integer and keep the index in order to be able to backtrack later to the original data. An example of such mapping is the following:

Example. Assume that the size of the monitoring window is 4 transactions and the set of literals in the database is *{a, b, c, d}*. The corresponding set of extended items and their integer mapping are depicted in the table below.

Table 1. Integer mapping example

Set of Extended Items	a_0	a_1	a_2	a_3	b_0	b_1	b_2	b_3	c_0	c_1	c_2	c_3	d_0	d_1	d_2	d_3
Integer Mapping:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

The following step is the mining of the frequent itemsets from the transformed data. For this purpose we use FP-tree [10], a very efficient, state-of-the-art algorithm. FP-tree (stands for "Frequent Pattern-tree") is an extended prefix tree structure that stores crucial information about frequent patterns. Its major advantage is that it reduces the number of database scans to only 2 in order to construct the FP-tree and produce the frequent itemsets, while other algorithms require a large number of repeated scans. Moreover, Apriori-based algorithms are generally inefficient when the number of different items is large, which can easily be the case in the transformed database. We do not provide further information or examples of FP-tree algorithm, because it is not within the scope of this paper.

Summarizing, the steps of our algorithm are outlined below:

1. Move a sliding window across the transactions of the database until the next occurrence of the target item is found. For every such occurrence, capture the corresponding monitoring window, transform it as described above and store it in a new database file. Store the integer-mapping index.
2. Mine the transformed database for frequent itemsets, using the FP-tree algorithm.
3. Given the fact that the target event occurs w time points after the monitoring window, produce the rules from the patterns mined in the previous step.

4. Using the integer-mapping index, convert the items of the rules, from integer numbers into their original form.

Algorithm analysis and discussion

When speaking about computational complexity within the data mining context, what is mostly important is the number of database scans. When the main memory is not enough to fit the data, main memory based operations are insignificant compared to operations that require hard disk access. The major advantage of our algorithm is that it requires only 3 scans over the database, regardless of the size of the database or the number of literals; one for the first step (sliding window) and two for the construction of the FP-tree [10].

Moreover, the main memory structures used are small and pose no additional overhead. The size of the sliding window is $m \times MaxTransactionSize$, and the size of the integer mapping index is (Monitoring Window Size) x (Size of the set of extended items), both of which are appropriate for the main memory. The largest main memory structure used is the FP-tree, a kind of prefix tree. It has the benefit that it is a highly compact structure whose size is bounded by the size of the database. Each transactions contributes at most one path to the FP-tree, with length equal to the number of frequent items in that transaction. Since there are often a lot of sharing of frequent items among transactions, the size of the tree is usually much smaller than its original database and that of the candidate sets generated in the Apriori-based approaches.

5. Implementation and Performance Results

We implemented our algorithm in C and tested it against a number of data sets of different sizes. The datasets were created using a MATLAB routine, according to a set of probabilistic pseudo-random parameters, such as the frequency of the rare event, the Monitoring Time and the Warning. The performance of the algorithm depends on the size of the monitoring window, the number of different items and the frequency of the target event. Below, we present an experimental setup that has the following configuration: There are eleven different items in the database, including the target item. The Monitoring Time was set to ten and the Warning Time to five, which means that, according to the m-w+1 formula the size of the Monitoring Window was six. Therefore, the transformed database contains 66 different extended items. The target event frequency was set between 9% and 10%, therefore, the transformed database contained approximately $0.1 \times DatabaseSize$ Mega Transactions. The experiments were taken on a Pentium 3, 1GHz computer with 512MB of RAM and a SCSI hard disk. The results of the experimental setup described above are summarized in Table 2. Figure 3 illustrates the performance of our algorithm with respect to the number of Mega Transactions of the transformed database. The times shown below do not include the run time of the frequent itemset mining algorithm we used (FPT), since any such algorithm can be used, are indicative however of the efficiency of our algorithm. More information about FPT can be found in [10].

Table 2. Experimental Results

DB Size (in transactions)	Mega Transactions	Run Time (seconds)
5000	479	5.1
10000	962	14.3
50000	4885	158
100000	9424	995

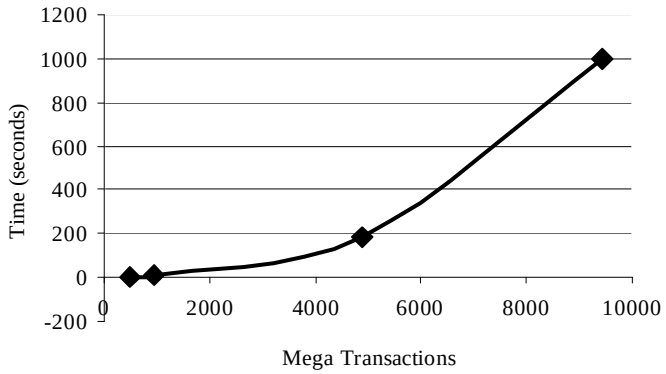

Fig. 3. Run time against the number of Mega Transactions

In all cases, the expected rules predicting the target event were successfully discovered. Our approach is complete due to the completeness of FPT. In the setup of our experiments, a frequent extended itemset starting from time point 0 and ending at time point 4 can predict an event that will happen at time point 11. For example, given that the target item is *x*, one of the discovered frequent itemsets, such as $\{a_0, b_1, b_2, d_4\}$ can produce the following rule:

IF (a at t_0) AND (b at t_1) AND (b at t_2) AND (d at t_4)
THEN x will happen at t_{11}

The candidate rules produced from the frequent extended itemsets are required to satisfy the minimum confidence criterion set by the user. The confidence of the rule is calculated as defined in formula (2) in paragraph 3.

6. Conclusions and Further Research

In this paper we proposed a novel data mining approach for predicting rare events in transaction databases. Our approach is based on the inter-transactional association rules framework and utilizes a state-of-the-art algorithm for classical association rules

mining, namely FP-Tree, in order to produce predictive rules. It involves a database transformation in order to extract only the required information before mining for the predictive rules. We formulated the problem, proposed a novel algorithm and conducted experiments to test and verify its performance.

The proposed algorithm features low computational cost, since it requires only three scans over the database. Additionally it is also memory efficient as it uses small memory based structures, except from the FP-tree, which in extreme cases can be relatively large. However, the FP-tree approach is a well established, complete and one of the most efficient approaches in the literature, which is the reason we selected it. The overall approach is complete due to the completeness of the frequent itemset algorithm.

For future research we consider conducting a large series of real world experiments and finally, extend our approach for distributed databases, such as web databases.

References

1. Beyond Intra-transaction Association Analysis: Mining Multi-dimensional Inter-transaction Association Rules (Lu, Feng, Han, ACM journal, 30 pages)
2. Breaking the barrier of transactions: Mining Inter-Transaction Association Rules. Anthony K. H. Tung, Hongjun Lu, Jiawei Han, Ling Feng, KDD '99.
3. Efficient Mining of Inter-transaction Association Rules (Tung, Lu, Han, Feng, TKDE January/February 2003)
4. Inter-transactional Association Rules for Multi-dimensional Contexts for Prediction and their Application to studying Meteorological Data.
5. Stock Movement Prediction and N-Dimensional Inter-transaction Association Rules (Lu, Han, Feng, extended abstract)
6. Rakesh Agrawal and Ramakrishnan Srikant. Mining Sequential Patterns. In Proc. of the 11th Int'l Conference on Data Engineering, Taipei, Taiwan, March 1995
7. R. Agrawal, M. Mehta, J. Shafer, R. Srikant, A. Arning, and T. Bollinger. The Quest data mining system. In Proc. 1996 Int. Conf. Data Mining and Knowledge Discovery (KDD'96), pages 244 -249, Portland, Oregon, August 1996.
8. H. Mannila and H. Toivonen and A. I. Verkamo. Discovering Frequent Episodes in Sequences. In Proc. First International Conference on Knowledge Discovery and Data Mining (KDD-95). AAAI Press. Montreal, Eds. U. M. Fayyad and R. Uthurusamy , Canada, 1995.
9. Gautam Das, King-Ip Lin, Heikki Mannila, Gopal Renganathan, and Padhraic Smyth. Rule Discovery from time series. In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98). AAAI Press, 1998.
10. J. Han, J. Pei, and Y. Yin, Mining Frequent Patterns without Candidate Generation, Proc. 2000 ACM-SIGMOD Int. Conf. on Management of Data (SIGMOD'00), Dallas, TX, May 2000.
11. P. Brockwell and R. Davis, Introduction to Time Series and Forecasting. Springer-Verlag New York, 1996.
12. Weiss, G. M., and Hirsh, H. 1998. Learning to Predict Rare Events in Event Sequences. In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, AAAI Press, 359-363.